Refine Search

Search Results -

Terms	Documents
L9 and L5	2

Database:

US Pre-Grant Publication Full-Text Database
[✓ US Patents Full-Text Database]
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L17

Refine Search

Recall Text

Clear

Interrupt

Search History

DATE: Monday, November 08, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=OR			
L17	L9 and l5	2	L17
L16	L9 and l4	38	L16
L15	L14 same match$	12	L15
L14	L10 same judg$	118	L14
L13	L10 and l9	2	L13
L12	L9 and l5	2	L12
L11	L10 and l5	1	L11
L10	L7 same l8	4370	L10
L9	multilevel adj1 memory adj1 cell	92	L9
L8	physical address	772632	L8
L7	logical adj1 address	4370	L7
L6	L5	94	L6
L5	L4 same l3 same l2	94	L5
L4	output adj1 voltage	118141	L4

L3	reference adj1 voltage	71985	L3
L2	msb or (most adj1 significant adj1 bit)	30478	L2
L1	6023781[pn]	1	L1

END OF SEARCH HISTORY

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L13: Entry 1 of 2 File: USPT Jul 10, 2001

DOCUMENT-IDENTIFIER: US 6260172 B1
TITLE: Semiconductor device with logic rewriting and security protection function

Brief Summary Text (38):
A semiconductor device with security protection function, the semiconductor device having logic means capable of changing logic thereof in response to a command signal corresponding to at least either a ciphering or a deciphering program, the logic means processing a signal with the rewritten logic, the logic means may comprise: a logic integrated circuit capable of changing logic thereof responsive to a command signal; and prohibiting means for comparing the command signal and at least a reference signal to prohibit rewriting of the logic of the logic integrated circuit when the command signal and the reference signal are identical to each other. The semiconductor device may further comprise memory means for storing a plurality of reference signals, the prohibiting means comparing the command signal with the plurality of reference signals. The logic integrated circuit may include: a plurality of calculating means each capable of a specific calculation; and wiring means for changing wiring among the plurality of calculators in response to the command signal to rewrite the logic of the logic integrated circuit. The logic integrated circuit may include a field programmable gate array. The logic integrated circuit may include a plurality of multilevel memory cells, each cell having a control gate and a floating gate. The logic integrated circuit may include a plurality of multilevel memory cells of at least a member of the group consisting of an MNOS, a mask ROM, an EEPROM, an EPROM, a PROM, a FRAM and a non-volatile flash memory. The semiconductor device may further comprise means for changing the logic of the logic integrated circuit. The semiconductor device may further comprise means for reading the logic of the logic integrated circuit. The semiconductor device may be at least either a contact type or non-contact type. The semiconductor device may further comprise antenna means for communications of the signal with an external unit by means of an electromagnetic wave.

Brief Summary Text (39):
A method of logic rewriting may be used for a semiconductor device having a logic-rewritable logic circuit, logic thereof being rewritable in response to an external command signal, the logic circuit being provided with a plurality of multilevel memory cells storing at least three levels of data each. The method may comprise the steps of: comparing a first data related to a first logical state indicating a present state of the logic circuit and a second data related to a second logical state of the logic circuit included in the command signal; prohibiting rewriting of the logic of the logic circuit from the first to the second logical state when the first and second data are not identical to each other; when the first and second data are identical to each other, inputting a first code composed of a plurality of first data bits and a second code composed of a plurality of second data bits, the first and second codes having been coded by a coding method; arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; generating a plurality of voltages corresponding to the arranged bits; and applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells to rewrite the logic of the logic circuit from the first to the second logical state.

Brief Summary Text (40):
Further, the present invention provides a method of changing logic, comprising the steps of: comparing an external command signal and at least one reference signal, the command signal being applied to a logic integrated circuit capable of changing logic thereof in response to the command signal and having a plurality of multilevel memory cells, each cell storing at least three levels of data each; prohibiting rewriting of the logic of the logic integrated circuit when the command signal and the reference signal are identical to each other; when the command signal and the reference signal are not identical to each other, in response to a first code composed of a plurality of first data bits and a second code composed of a plurality of second data bits, the first and second codes having been coded by a coding method, arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; generating a plurality of voltages corresponding to the arranged bits; and applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells.

Brief Summary Text (41):
A computer readable medium may store program code for causing a computer to apply a program for rewriting logic of a logic-rewritable logic circuit in response to an external command signal, the logic circuit being provided with a plurality of multilevel memory cells storing at least three levels of data each. The program code may comprise: first program code means comparing a first data related to a first logical state indicating a present state of the logic circuit and a second data related to a second logical state of the logic circuit included in the command signal; second program code means for prohibiting rewriting of the logic of the logic circuit from the first to the second logical state when the first and second data are not identical to each other; third program code means for, when the first and second data are identical to each other, inputting a first code composed of a plurality of first data bits and a second code composed of a plurality of second data bits, the first and second codes having been coded by a coding method; fourth program code means for arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; fifth program code means for generating a plurality of voltages corresponding to the arranged bits; and sixth program code means for applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells to rewrite the logic of the logic circuit from the first to the second logical state.

Brief Summary Text (42):
A computer readable medium storing program code for causing a computer to apply a program for preventing logic of a logic integrated circuit capable of changing logic thereof in response to an external command signal, the logic integrated circuit having a plurality of multilevel memory cells, each cell storing at least three levels of data each, the program may comprise: first program code means for comparing the command signal and at least a reference signal; second program code means for prohibiting rewriting of the logic of the logic integrated circuit when the command signal and the reference signal are identical to each other; third program code means, when the command signal and the reference signal are not identical to each other, for inputting at least a first code composed of a plurality of first data bits and a second code composed of a plurality of first data bits, the first and second codes having been coded by a coding method; fourth program code means for arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; generating a plurality of voltages corresponding to the arranged data bits; and applying the voltages to cells among the plurality of multilevel memory cells in

response to address information corresponding to the cells.

Brief Summary Text (43):
A code processing method may be used for a semiconductor device having a logic-rewritable logic circuit with a logic security function and having a plurality of multilevel memory cells storing at least three levels of data each. The method may comprise the steps of: supplying a code processing program to the control device; outputting a command signal corresponding to the program from the control device; applying the command signal to the logic circuit to rewriting logic thereof in accordance with the program; applying the program to at least either data or an address related to the data on the basis of the rewritten logic; and storing the data in a memory device, wherein the logic rewriting step includes the steps of: comparing a first data related to a first logical state indicating a present state of the logic circuit with a second data related to a second logical state of the logic circuit included in the command signal; prohibiting rewriting of the logic of the logic circuit from the first to the second logical state when the first and the second data are not identical to each other; when the first and the second data are identical to each other, inputting at least a first code composed of a plurality of first data bits and a second code composed of a plurality of first data bits, the first and second codes having been coded by a coding method; fourth program code means for arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; generating a plurality of voltages corresponding to the arranged data bits; and applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells to rewrite the logic of the logic circuit from the first to the second logical state.

Brief Summary Text (44):
A code processing method may be used for a semiconductor device having a logic-rewritable logic circuit with a logic security function and having a plurality of multilevel memory cells storing at least three levels of data each. The method may comprise the steps of: supplying a code processing program to the control device; outputting a command signal corresponding to the program from the control device; applying the command signal to the logic circuit to rewriting logic thereof in accordance with the program; applying the program to at least either data or an address related to the data on the basis of the rewritten logic; and storing the data in a memory device, wherein the logic rewriting step includes the steps of: comparing the command signal with at least a reference signal; prohibiting rewriting of the logic of the logic circuit when the command signal and the reference signal are identical to each other; when the command signal and the reference signal are not identical to each other, inputting at least a first code composed of a plurality of first data bits and a second code composed of a plurality of first data bits, the first and second codes having been coded by a coding method; arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; generating a plurality of voltages corresponding to the arranged data bits; and applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells to rewrite the logic of the logic circuit.

Brief Summary Text (45):
A computer readable medium may storage program code for causing a computer to apply a program for code processing of data of a semiconductor device with a security protection function having a logic integrated circuit capable of changing logic thereof in response to an external command signal, the logic integrated circuit having a plurality of multilevel memory cells storing at least three levels of data each, and a control device. The program may comprise: first program code means for supplying a code processing program to the control device; second program code

means for outputting a command signal corresponding to the program from the control device; third program code means for applying the command signal to the logic circuit to rewriting logic thereof in accordance with the program; fourth program code means for applying the program to at least either data or an address related to the data on the basis of the rewritten logic; and fifth program code means for storing the data in a memory device, wherein the third program code means includes: sixth program code means for comparing a first data related to a first logical state indicating a present state of the logic circuit with a second data related to a second logical state of the logic circuit included in the command signal; seventh program code means for prohibiting rewriting of the logic of the logic circuit from the first to the second logical state when the first and the second data are not identical to each other; eighth program code means for, when the first and the second data are identical to each other, inputting at least a first code composed of a plurality of first data bits and a second code composed of a plurality of first data bits, the first and second codes having been coded by a coding method; ninth program code means for arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; tenth program code means for generating a plurality of voltages corresponding to the arranged data bits; and eleventh program code means for applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells to rewrite the logic of the logic circuit from the first to the second logical state.

Brief Summary Text (46):
A computer readable medium may storage program code for causing a computer to apply a program for code processing of data of a semiconductor device with a security protection function having a logic integrated circuit capable of changing logic thereof in response to an external command signal, the logic integrated circuit having a plurality of multilevel memory cells storing at least three levels of data each, and a control device. The program may comprise: first program code means for supplying a code processing program to the control device; second program code means for outputting a command signal corresponding to the program from the control device; third program code means for applying the command signal to the logic circuit to rewriting logic thereof in accordance with the program; fourth program code means for applying the program to at least either data or an address related to the data on the basis of the rewritten logic; fifth program code means for storing the data in a memory device, wherein the third program code means includes: sixth program code means for comparing the command signal with at least a reference signal; seventh program code means for prohibiting rewriting of the logic of the logic circuit when the command signal and the reference signal are identical to each other; eighth program code means for, when the command signal and the reference signal are not identical to each other, inputting at least a first code composed of a plurality of first data bits and a second code composed of a plurality of first data bits, the first and second codes having been coded by a coding method; ninth program code means for arranging the first and the second data bits in order that at least a bit of an N-order of the first data bits and a bit of the N-order of the second data bits are stored in one of the cells, the N being an integral number; tenth program code means for generating a plurality of voltages corresponding to the arranged data bits; and eleventh program code means for applying the voltages to cells among the plurality of multilevel memory cells in response to address information corresponding to the cells to rewrite the logic of the logic circuit.

Detailed Description Text (125):
In the embodiments described above, EEPROM memory cells are used as the memory cells 202.sub.x0 202.sub.xm. Not limited to this, another memory cell, such as multilevel memory cell composed of a ferroelectric random access memory (FRAM) described in Japanese Patent Laid-Open No. 8(1996)-124378 can be used.

Detailed Description Text (127):
Further, as the memory cells 202.sub.x0 to 202.sub.xm, other multilevel memory cells of MNOS, mask ROM, RPROM, PROM and flash non-volatile memory can be used.

Detailed Description Text (146):
A method of deciphering secret data stored in the mask ROM 501a can be achieved with the FPGA 502 to convert a logical address AD-1 used for the CPU 503 into a structural address AD-2 used for the mask ROM 501a. Here, the structural address AD-2 is used to decipher and further read a lump of data ciphered and stored at scattered (at random) positions in the mask ROM 501a. In other words, as described later, the FPGA 502 receives the logical address AD-1 from the CPU 503, converts the logical address AD-1 into the structural address AD-2, and outputs the converted addresses to the mask ROM 501a.

Detailed Description Text (154):
Upon end of this programming, in step S705, the CPU 503 transmits a signal indicative of that data can be output, to the external apparatus 600 via data I/O section 504. In response to the signal, data of a logical address is requested to be output from the external apparatus 600. In step S706, the CPU 503 receives the logical address AD-1 via data I/O section 504 and outputs the address to the FPGA 502.

Detailed Description Text (155):
In step S707, the FPGA 502 calculates the structural address AD-2 on the mask ROM 501a on the basis of the logical address AD-1 in accordance with the program set in the step S704. The calculated structural address AD-2 is output to the mask ROM 501a. Further, in step S708, the mask ROM 501a outputs data D to the CPU 503 on the basis of the structural address AD-2 calculated by the FPGA 502. The CPU 503 then outputs the data D to the external apparatus 600 via data I/O section 504.

Detailed Description Text (162):
Upon end of this programming, in step S714, the CPU 503 transmits a signal indicative of that data can be input, to the external apparatus 600 via data I/O section 504. In response to the signal, data of a logical address is requested to be input from the external apparatus 600. In step S715, the CPU 503 receives the logical address AD-1 via data I/O section 504, the CPU 503 outputs the received logical address AD-1 to the FPGA 502.

Detailed Description Text (163):
In step S716, the FPGA 502 calculates the structural address AD-2 on the EEPROM 501b on the basis of the logical address AD-1 in accordance with the program set in the step S704, and outputs the calculated structural address to the EEPROM 501b. Further, in step S717, data is input from the external apparatus 600 to the CPU 503 via data I/O section 504, and then stored as the structural AD-2 to the EEPROM 501b. When a user wants to change a cipher key, it can be stored in the FPGA 502.

Detailed Description Text (169):
In order to decipher the data ciphered and then stored in the memory device 600 (i.e., processing in step S707 of FIG. 17), the structural address AD-2 is calculated that is shifted to the left from the logical address AD-1 input from the CPU 503 to the FPGA 502, and further the program for deciphering the address is written in the FPGA 502. Then, it is possible to read the data deciphered in accordance with this program.

Detailed Description Text (196):
In this case, in the fourth embodiment, for instance, secret data output from the CPU 503 to the external apparatus 600 via I/O section 504 can be ciphered as follows: the CPU 503 sets a program for interpreting and ciphering data output from the external apparatus 600 to the FPGA 502. Further, the FPGA 502 converts the logical address AD-1 supplied from the CPU 503 into the structural address AD-2.

CLAIMS:

5. The semiconductor device of claim 3, wherein the logic integrated circuit includes a plurality of multilevel memory cells each storing one of at least three memory states, each cell having a control gate and a floating gate.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

☐ Generate Collection Print

L12: Entry 1 of 2 File: USPT Jan 6, 2004

DOCUMENT-IDENTIFIER: US 6674385 B2
TITLE: Analog-to-digital conversion method and device, in high-density multilevel non-volatile memory devices

Abstract Text (1):

An analog-to-digital conversion method and device for a multilevel non-volatile memory device that includes a multilevel memory cell. The method comprises a first step of converting the most significant bits contained in the memory cell, followed by a second step of converting the least significant bits. The first step is completed within a time interval corresponding to the rise transient of the gate voltage, and the second step is initiated at the end of the transient. Also disclosed is a scheme for error control coding in multilevel Flash memories. The n bits stored in a single memory cell are organized in different "bit-layers", which are independent from one another. Error correction is carried out separately for each bit-layer. The correction of any failure in a single memory cell is achieved by using a simple error control code providing single-bit correction, regardless of the number of bits stored in a single cell.

Brief Summary Text (3):

This invention relates to an analog-to-digital conversion method in high-density multilevel non-volatile memory devices, being of the type wherein a reading operation is performed on a multilevel memory cell, comprising a floating gate transistor with drain and source terminals, by applying predetermined bias voltage values to its drain and source terminals while its drain terminal is applied a predetermined current value, and by measuring the value of its gate voltage.

Brief Summary Text (13):

By contrast, a multilevel memory cell can store a larger number of bits than one. From the electrical standpoint, this means that the threshold voltage can have more than two values. The amount of information that can be stored in a single multilevel cell increases according to the following relation:

Brief Summary Text (14):

From a physical standpoint, the ability to alter the threshold voltage Vth, and hence to program the multilevel memory cell, is afforded by the floating gate structure of the transistor which comprises the memory cell. The gate region is isolated in D.C. but accessible through charge injection processes of the Channel Hot Electrons and/or the Fowler-Nordheim Tunneling Effect types.

Brief Summary Text (53):

The invention relates to an analog-to-digital conversion method in high-density multilevel non-volatile memory devices, comprising the following steps: reading a multilevel memory cell, comprising a floating gate transistor with drain and source terminals, by applying predetermined bias voltage values to its drain and source terminals while its drain terminal receives a predetermined current value; measuring the value of the cell gate voltage; converting the most significant bits (MSB) contained in the memory cell; converting the least significant bits (LSB) contained in the memory cell.

Brief Summary Text (54):

The invention further relates to an analog-to-digital conversion device incorporated in high-density multilevel non-volatile memory devices, of the type used for reading the contents of a multilevel memory cell comprising a floating gate transistor with drain and source terminals, and further comprising a plurality of voltage comparators, each having a first input coupled to the floating gate and a second input maintained at a corresponding reference voltage value, the comparator outputs being connected to a logic block for extracting the most significant bits of the cell.

Drawing Description Text (9):
FIG. 8 shows schematically a set of multilevel memory cells which are programmed to reduce the time of latency in the synchronous read mode according to the inventive method; and

Detailed Description Text (3):
The method allows an analog-to-digital conversion to be carried out of a gate voltage Vg extracted during a step of voltage-mode reading from multilevel memory cells 12, as shown schematically in FIG. 8. Each cell has a large number of levels, e.g. eight to sixteen, and the conversion method allows the informational contents of the cell 12 to be discriminated within a very short time of approximately 100 ns. Thus, the total time of asynchronously accessing the memory can be kept at approximately 150 ns, and the operation be accompanied by a small amount of "read disturb" to the cells being read.

Detailed Description Text (64):
An embodiment of the invention provides an ECC scheme for multilevel Flash memories based on a binary code providing single-bit correction, which has a better correction capability than non-binary ECCs which correct any failure in a single cell. This code greatly simplifies the required encoding and decoding networks and minimizes the impact on memory access time. Moreover, the presented scheme can be applied to multilevel memories with cells working at a variable number of levels (i.e., with a variable number of bits stored per cell), thereby ensuring the required data protection in each operating mode by exploiting the same encoding and decoding circuits and check cells.

Detailed Description Text (87):
The error correction circuit 30 operates as follows when encoding information data being written (DATA_IN) into the memory. The DATA_IN is input into the multiplexer 32 which, in response to an appropriate control signal (not shown), passes the DATA_IN to the coder 36. The coder 36 generates appropriate check bits CHECK_IN from the DATA_IN using one of the error correction code schemes discussed above. The DATA_IN and CHECK_IN bits are then output to the program circuits that write the DATA_IN and CHECK_IN bits into a common layer of a set of multilevel memory cells.

CLAIMS:

1. An analog-to-digital conversion method in a high-density multilevel non-volatile memory device that includes a multilevel memory cell having a floating gate transistor with gate, drain, and source terminals, the method comprising: reading the memory cell by applying predetermined bias voltage values to the drain and source terminals while the drain terminal is applied a predetermined current value, and by detecting a gate voltage at the gate terminal; a first converting step of converting the gate voltage into a plurality of most significant bits; and a second converting step that follows the first converting step, the second converting step converting the gate voltage into a plurality of least significant bits.

7. An analog-to-digital conversion device incorporated in a high-density multilevel non-volatile memory device that includes a multilevel memory cell having a floating gate transistor with gate, drain, and source terminals, the device comprising: a

plurality of voltage comparators, each having a first input coupled to the gate terminal, a second input maintained at a corresponding reference voltage value, and an output, each voltage comparator being a time continual comparator; a logic block connected to the outputs of the voltage comparators and structured to extract most significant bits of the memory cell; and means for initiating analog-to-digital conversion of an analog voltage of the memory cell during a read transient phase.

9. An analog-to-digital conversion device incorporated in a high-density multilevel non-volatile memory device that includes a multilevel memory cell having a floating gate transistor with gate, drain, and source terminals, the device comprising: a plurality of voltage comparators, each having a first input coupled to the gate terminal, a second input maintained at a corresponding reference voltage value, and an output, each voltage comparator being a time continual comparator; a first logic block connected to the outputs of the voltage comparators and structured to extract most significant bits of the memory cell; and a selection block, connected downstream of the first logic block, effective to select additional reference voltage values for application to corresponding additional comparators provided for a subsequent second conversion step in order to extract least significant bits of the memory cell by means of another logic block being connected to outputs of the additional comparators.

Previous Doc Next Doc Go to Doc#